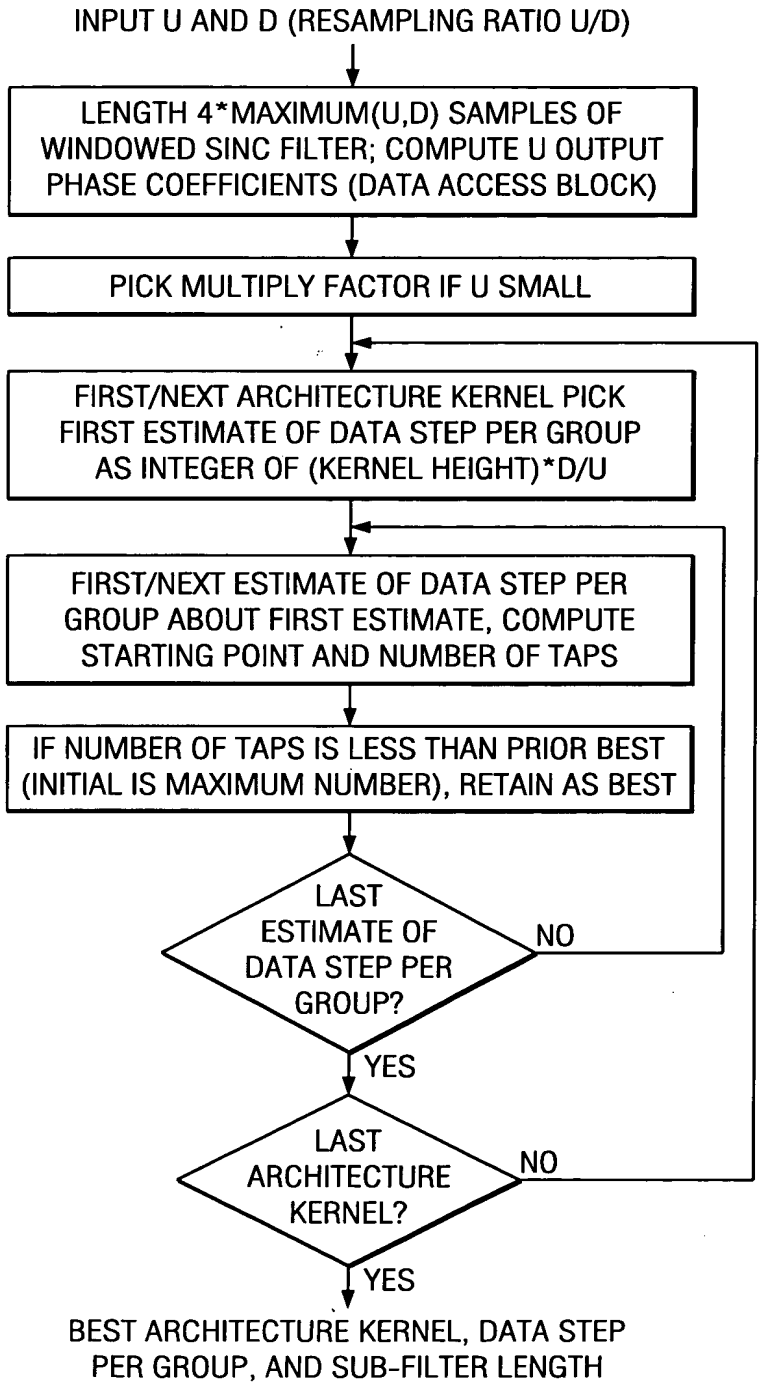


FIG. 1a

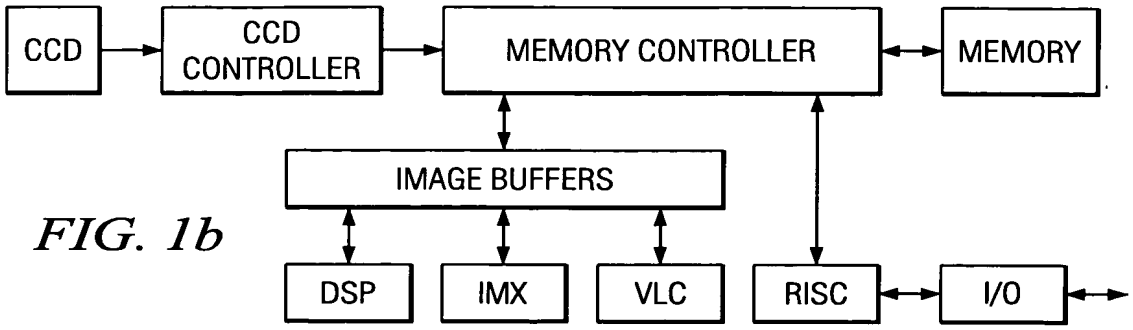


FIG. 1b

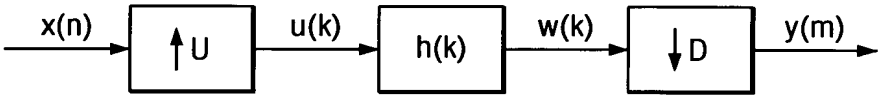


FIG. 2a

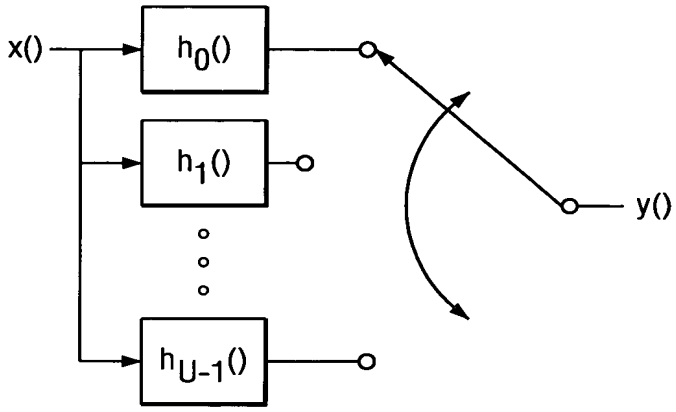


FIG. 2b

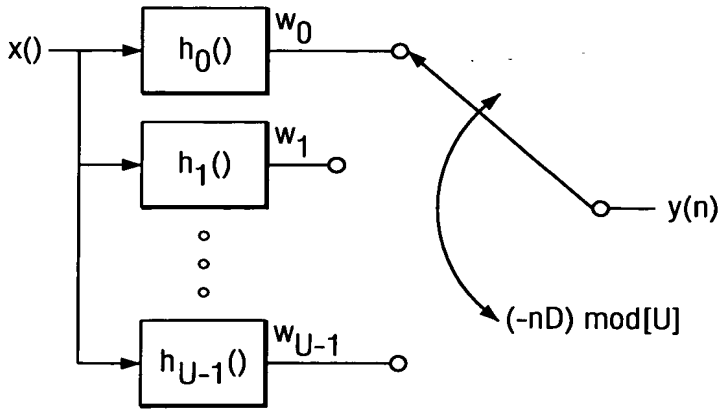


FIG. 2c

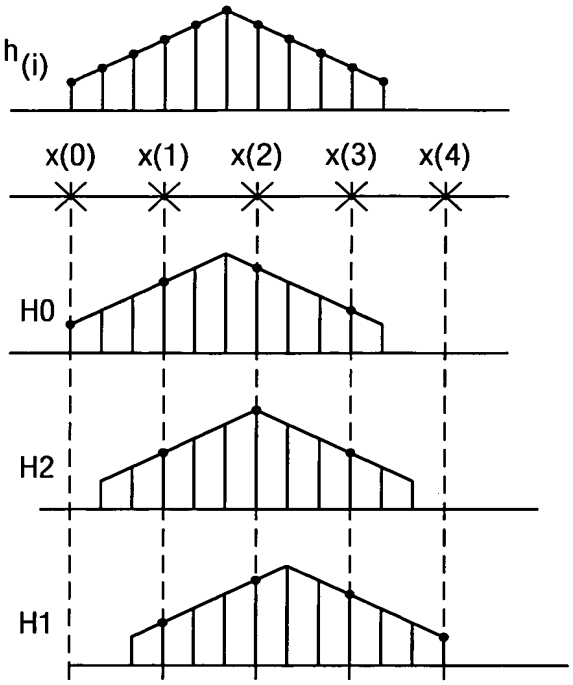


FIG. 3a

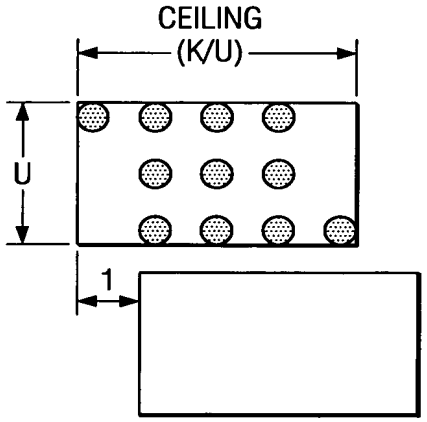


FIG. 3b

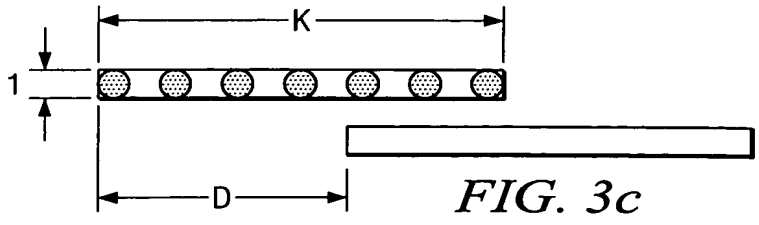


FIG. 3c

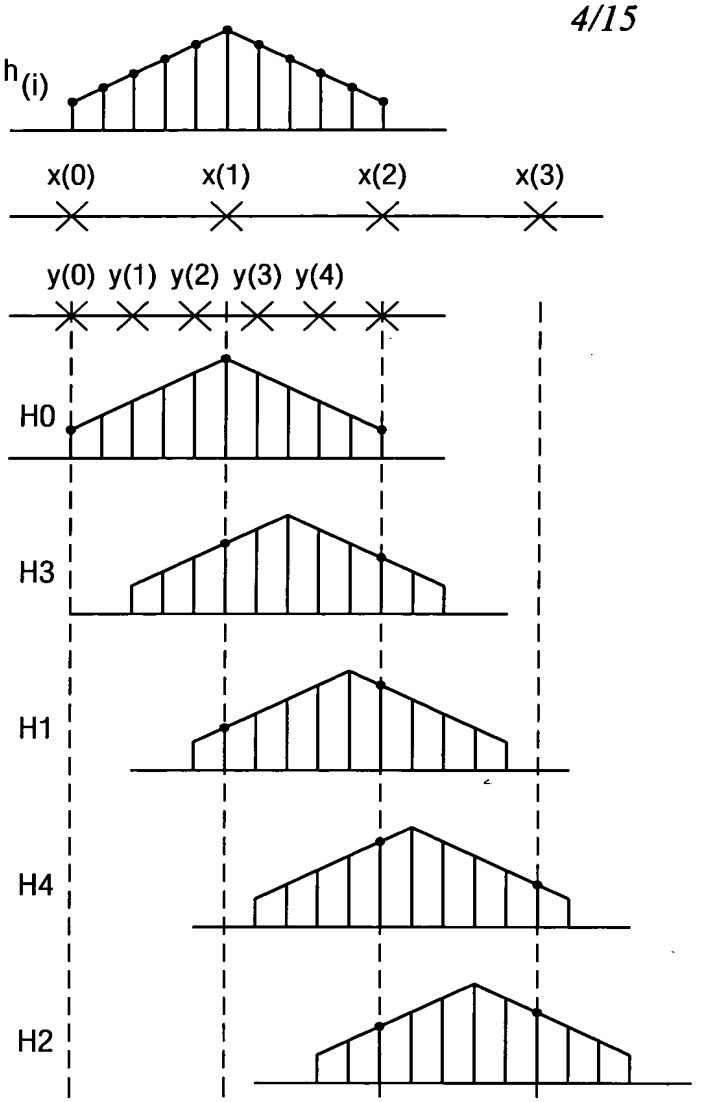


FIG. 4a

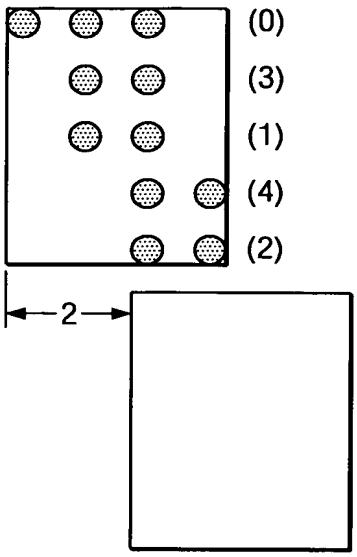


FIG. 4b

FIG. 11

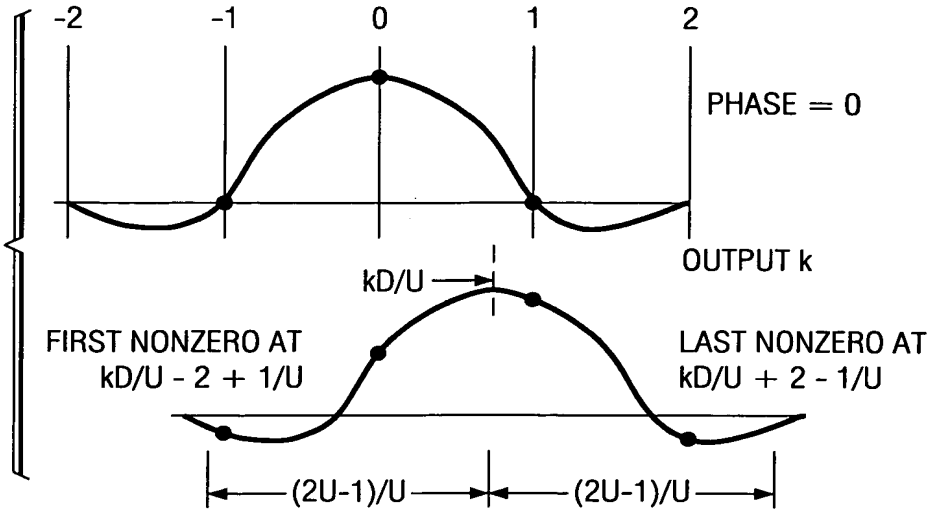
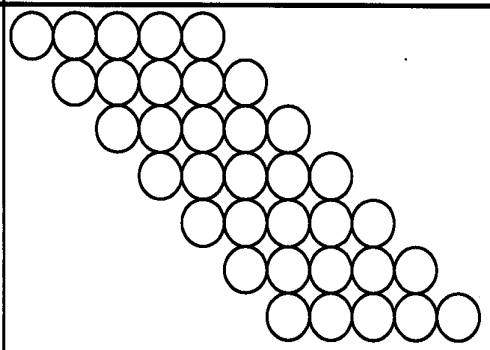
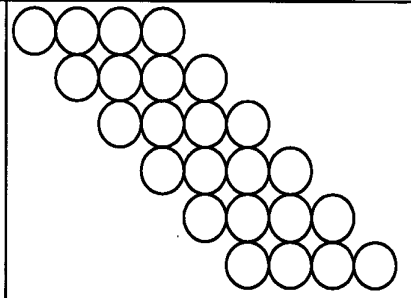


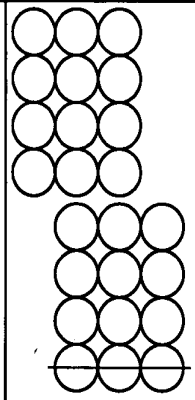
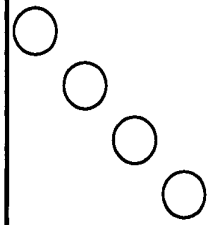
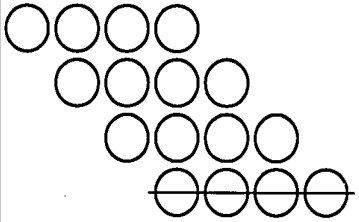
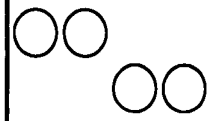
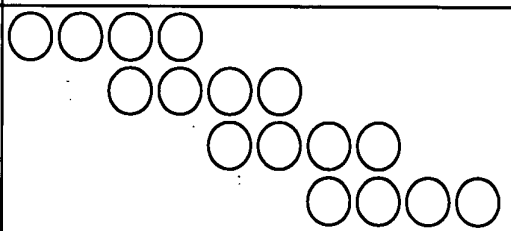
FIG. 5a

ARCHITECTURE KERNEL	EXPLANATION	EXAMPLE
SINGLE POINT 	SINGLE-THREAD FILTERING, GETTING ONE DATA POINT AT A TIME. THE NEXT ACCESS CAN STAY ON THE SAME POINT, OR MOVE FOR A FIXED DISTANT. THIS IS THE MOST FLEXIBLE PATTERN. HOWEVER, THERE IS USUALLY TIME ASSOCIATED WITH EACH LEVEL OF LOOPING, WHICH IS NEEDED TO IMPLEMENT CHANGING OF STEPPING DISTANCE IN A REGULAR MANNER. A PARALLEL DSP PERFORMING VERTICAL FILTERING OFTEN USE THIS SINGLE-POINT KERNEL, AS ITS PARALLELISM IS APPLIED TO DIFFERENT FILTERING PROBLEMS (FOR VERTICAL FILTERING, EACH COLUMN IS OPERATED INDEPENDENTLY AND IS THUS A SEPARATE PROBLEM)	 THIS IS FOR 5-TAP-PER-OUTPUT FILTERING GOING ON FOR 7 OUTPUTS, FOR ONE DATA ACCESS BLOCK FOR 7/D RESAMPLING
4-WIDE 	DSP WITH FIXED, 4-WIDE, DATA ACCESS, AND CAPABILITY TO COMPUTE INNER-PRODUCT WITH COEFFICIENT ARRAY AND PRODUCING A SINGLE SUM. TO BE EFFICIENT FOR FILTERING, THE STARTING POINT SHOULD BE ON ANY ALIGNMENT	 THIS IS 4-TAP-PER-OUTPUT FILTERING FOR A 6/D RESAMPLING

TO FIG. 5b

FIG. 5b

FROM FIG. 5a

4-TALL 	DSP WITH 4 PARALLEL EXECUTION UNITS, AND THEY ARE ALL FED WITH THE SAME SINGLE DATA POINT. THERE IS A SIGNIFICANT DISTINCTION IN DSP ARCHITECTURE THAT AFFECTS HOW WE CAN USE THIS AND MANY OTHER ARCHITECTURE KERNELS: WRITING IN ANY ALIGNMENT, OR WRITING ONLY ON 2^N-WORD ALIGNMENT. THE FORMER CAN BE USED TO IMPLEMENT ANY U FACTOR. THE LATTER CAN ONLY WORK WITH U BEING MULTIPLE OF 2^N. WHEN U IS NOT A MULTIPLE OF 4, FOR EXAMPLE, WE UPSCALE U AND D SO THAT THEY ARE, AT THE EXPENSE OF EFFICIENCY	 THIS IS A WRITE-ANY-ALIGNMENT ARCHITECTURE IMPLEMENTING A 7/D RESAMPLING WITH 3-TAP-PER-OUTPUT FILTERS. NOTE THAT 8 OUTPUTS ARE COMPUTED AND THEN ONE IS THROWN AWAY
1:1 SLOPE, 4 OUTPUTS 	DSP WITH 4 PARALLEL EXECUTION UNITS, AND THEY ARE FED WITH 4 DATA POINTS, ONE FOR EACH. TO BE EFFICIENT FOR FILTERING, THE 4 INPUTS NEED TO BE ON ANY WORD ALIGNMENT	 THIS IS FOR A 3/D RESAMPLING, WITH 4-TAP-PER-OUTPUT FILTERS. NOTE THAT 4 OUTPUTS ARE COMPUTED AND THEN ONE IS THROWN AWAY
1:2 SLOPE, 2 OUTPUTS 	DSP THAT CAN TAKE IN 4 INPUTS, AND PERFORM acc_A=acc_A+c0*d0+c1*d1, acc_B=acc_B+c2*d2+c3*d3 TO BE EFFICIENT FOR FILTERING, THE INPUTS NEED TO BE ON ANY WORD ALIGNMENT	 THIS IS FOR A 4/D RESAMPLING, WITH 4-TAP-PER-OUTPUT FILTERS

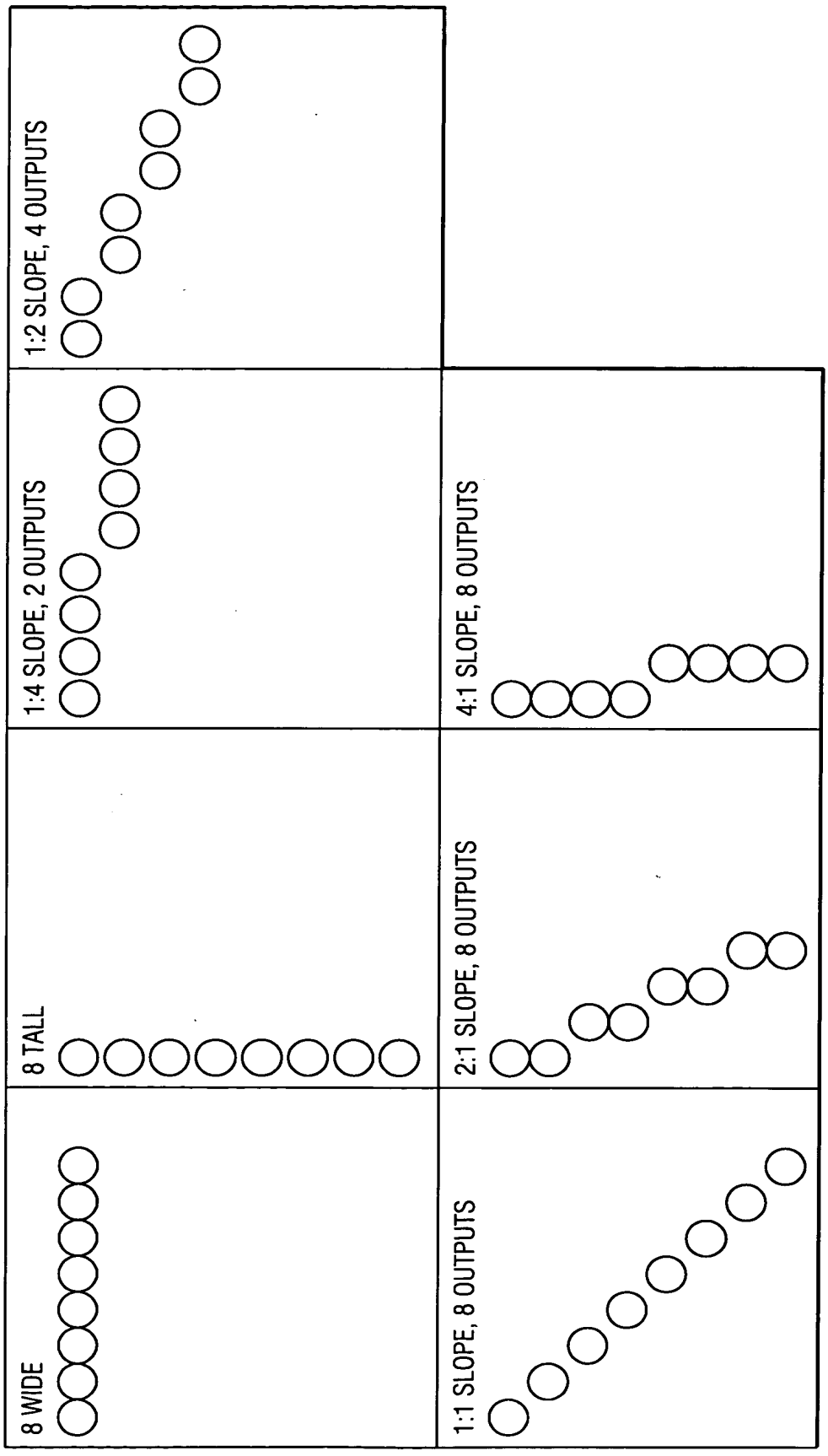

8 WIDE
8 TALL
1:4 SLOPE, 2 OUTPUTS
1:2 SLOPE, 4 OUTPUTS
1:1 SLOPE, 8 OUTPUTS
2:1 SLOPE, 8 OUTPUTS
4:1 SLOPE, 8 OUTPUTS

FIG. 6a

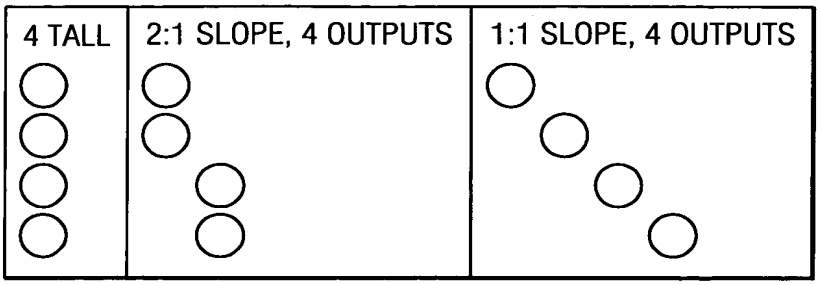


FIG. 6b

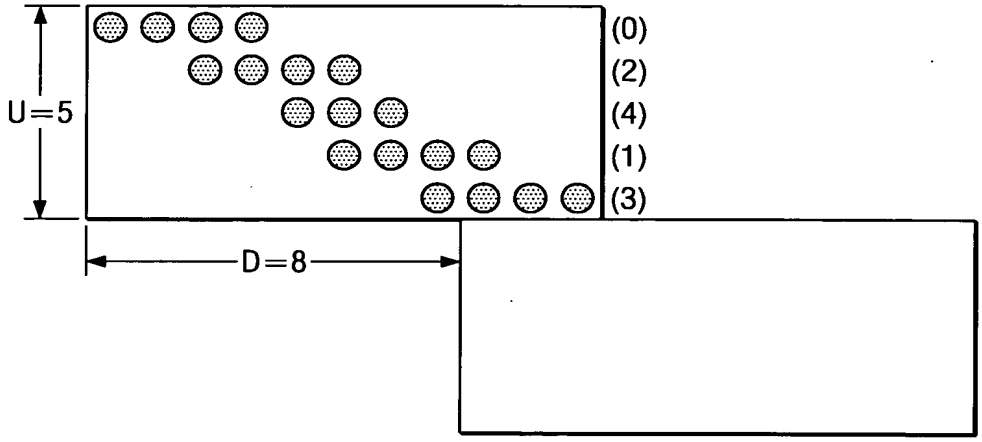


FIG. 7a

FIG. 7b

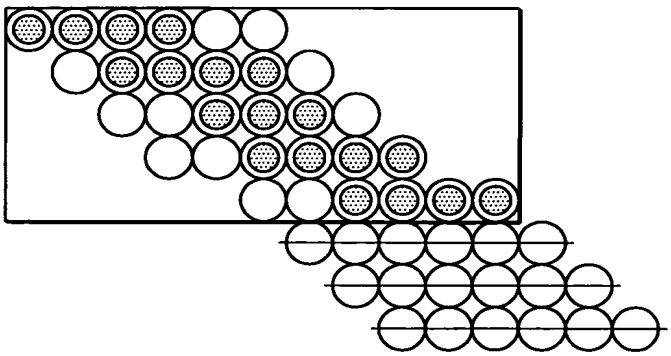
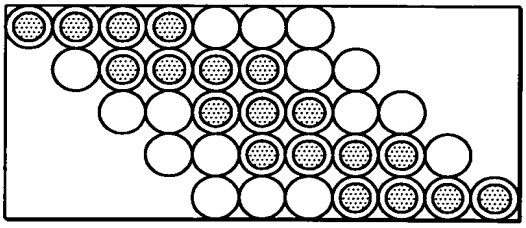


FIG. 7c



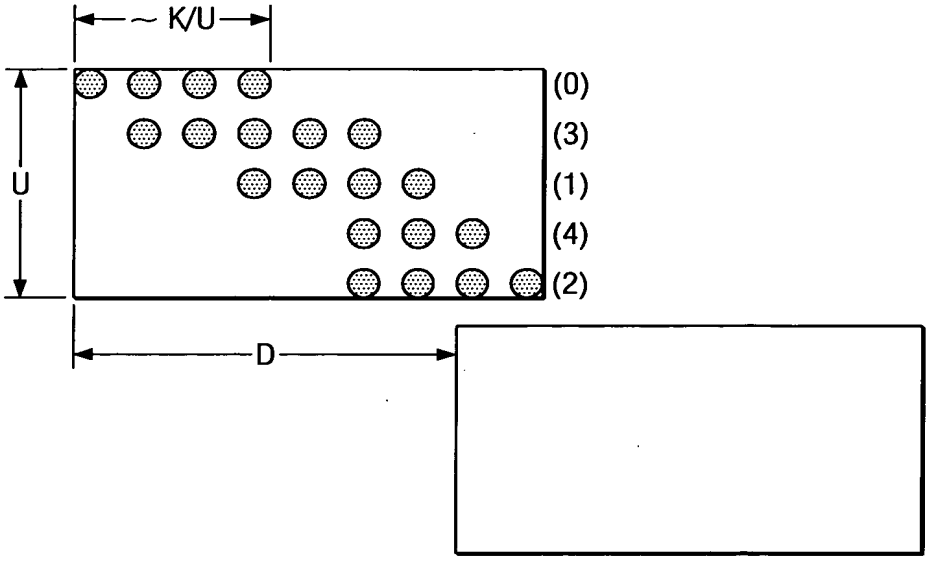


FIG. 8a

FIG. 8c

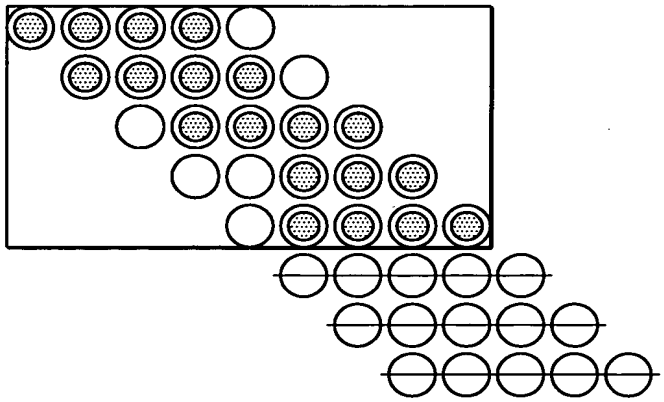
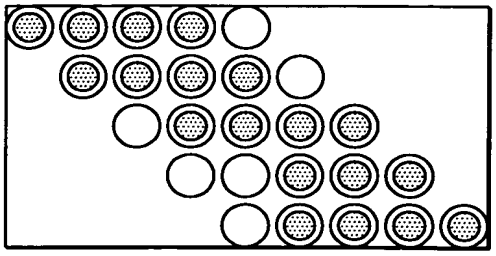


FIG. 8d



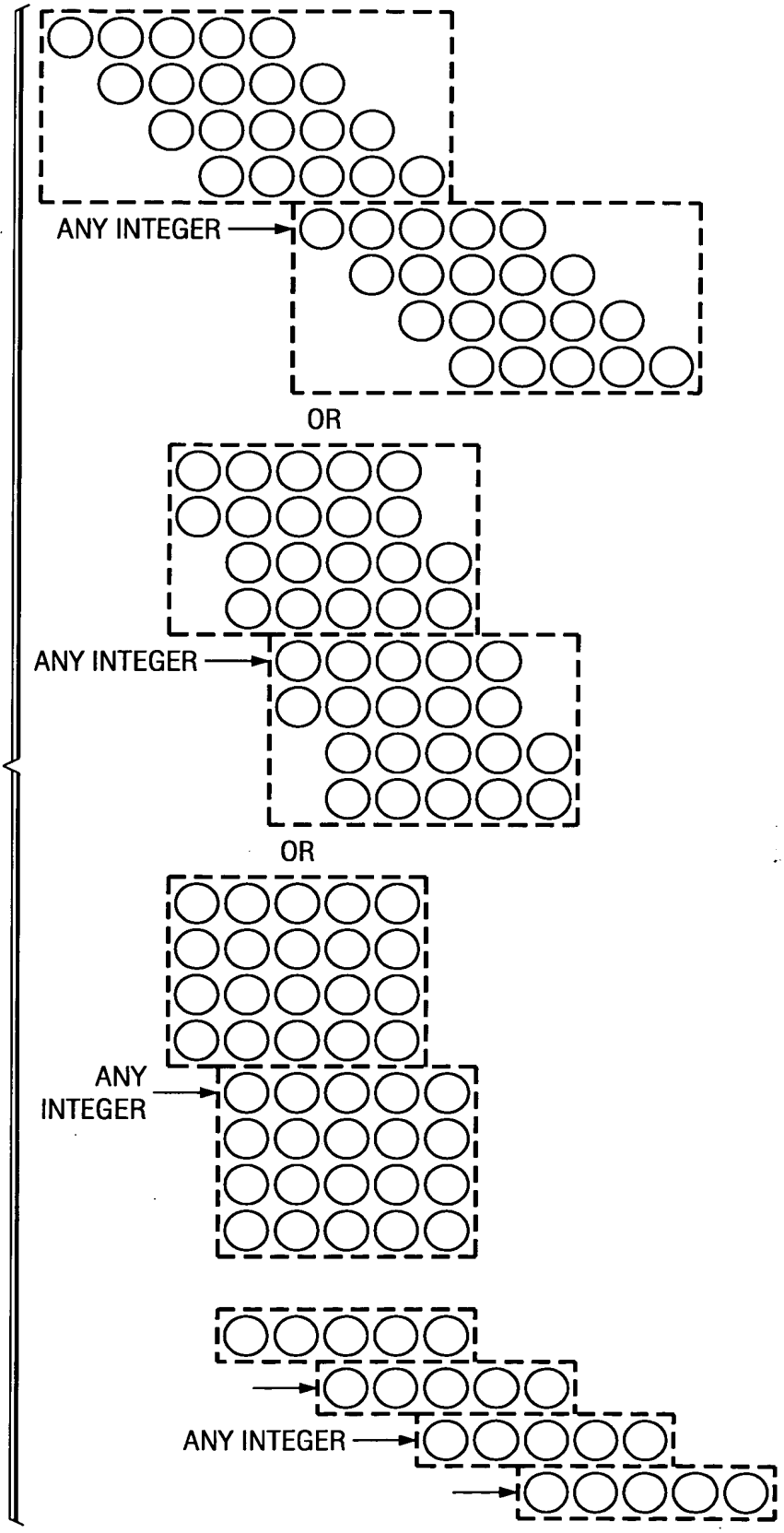


FIG. 8b

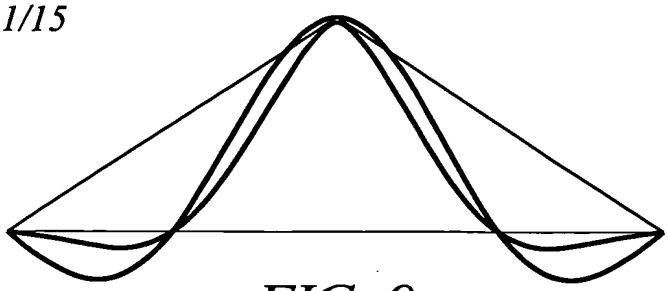


FIG. 9

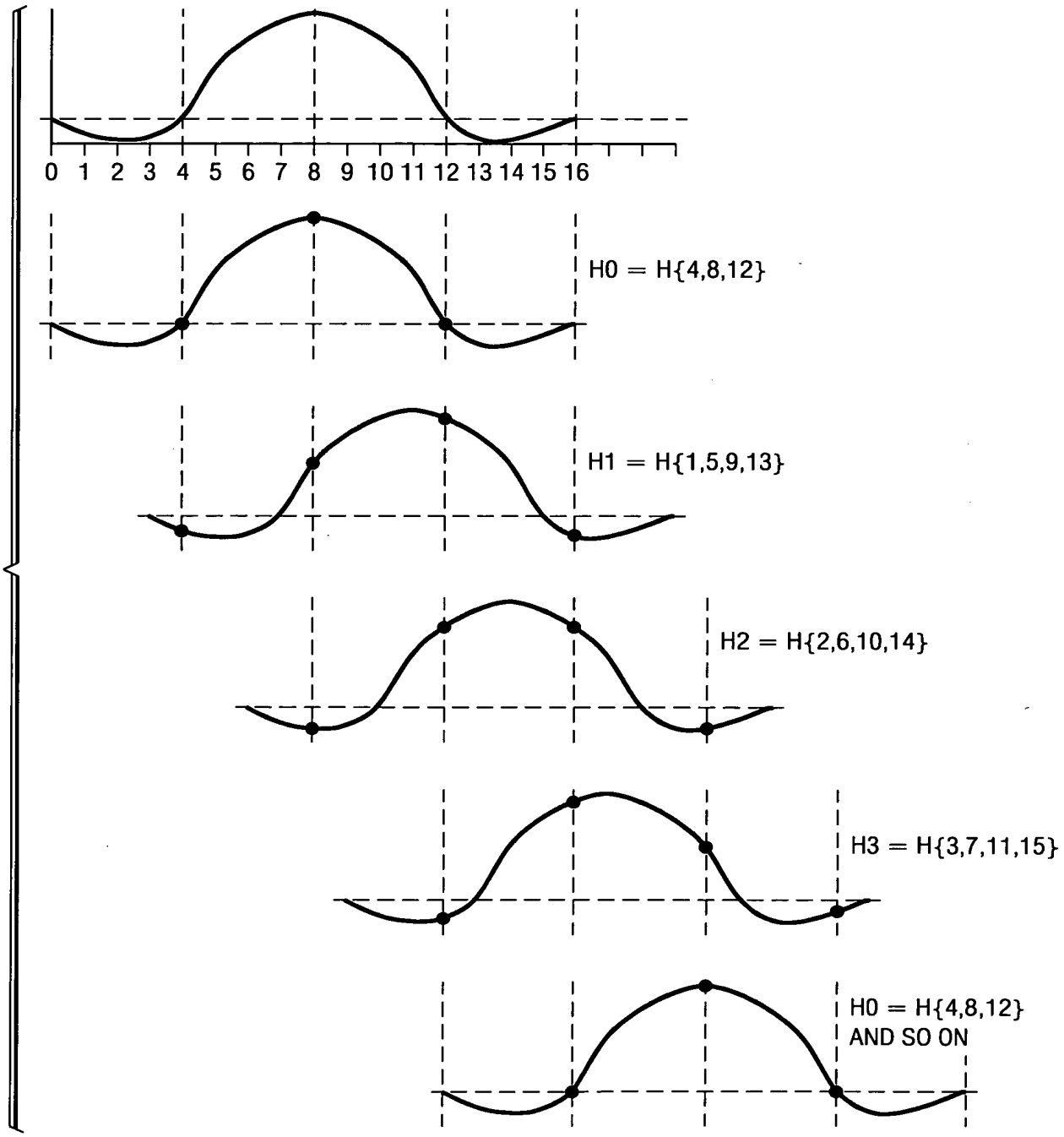

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
H0 = H{4,8,12}
H1 = H{1,5,9,13}
H2 = H{2,6,10,14}
H3 = H{3,7,11,15}
H0 = H{4,8,12}
AND SO ON

FIG. 10

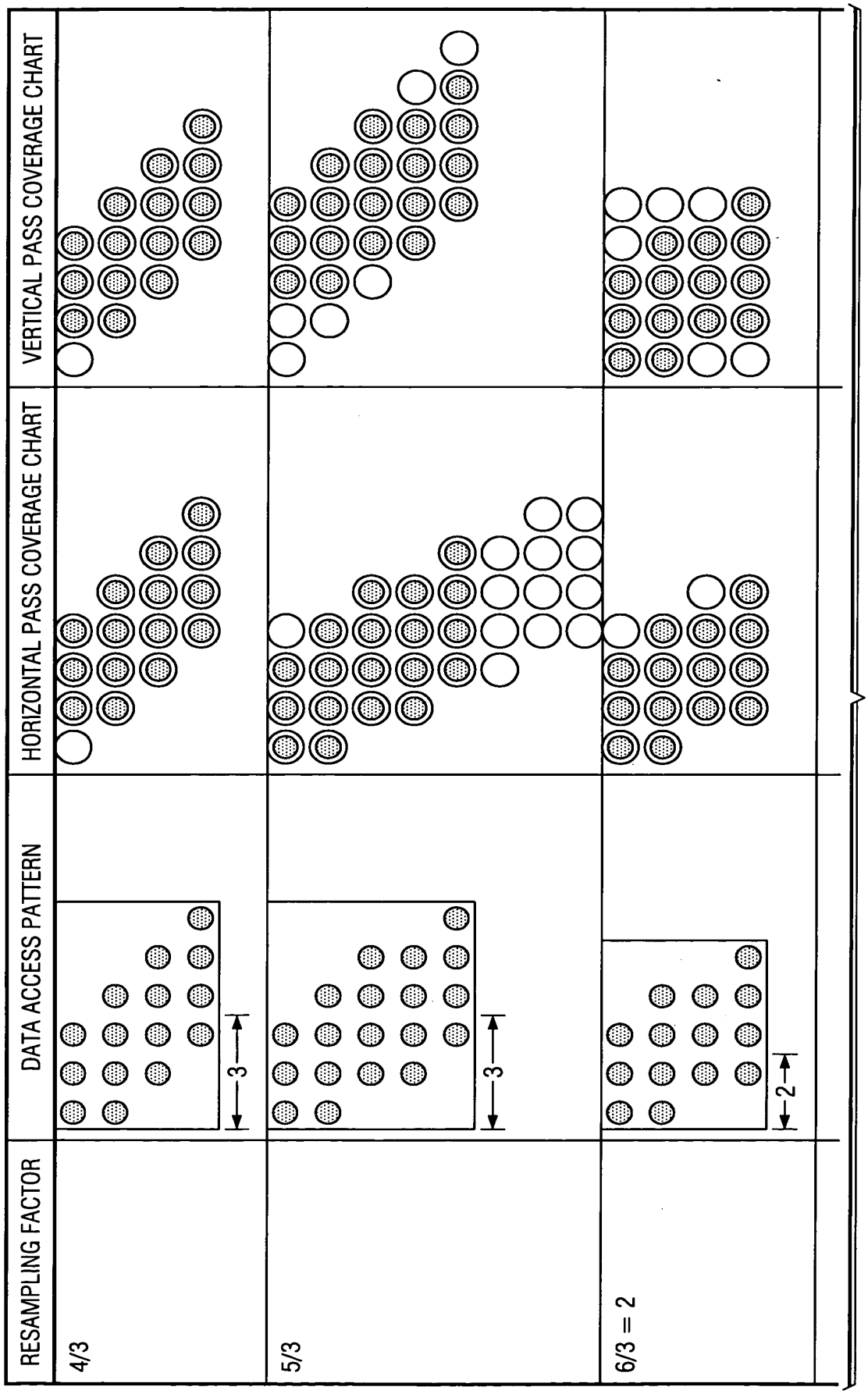

RESAMPLING FACTOR
DATA ACCESS PATTERN
HORIZONTAL PASS COVERAGE CHART
VERTICAL PASS COVERAGE CHART
4/3
3
5/3
3
6/3 = 2
2
TO FIG. 12b

FIG. 12a

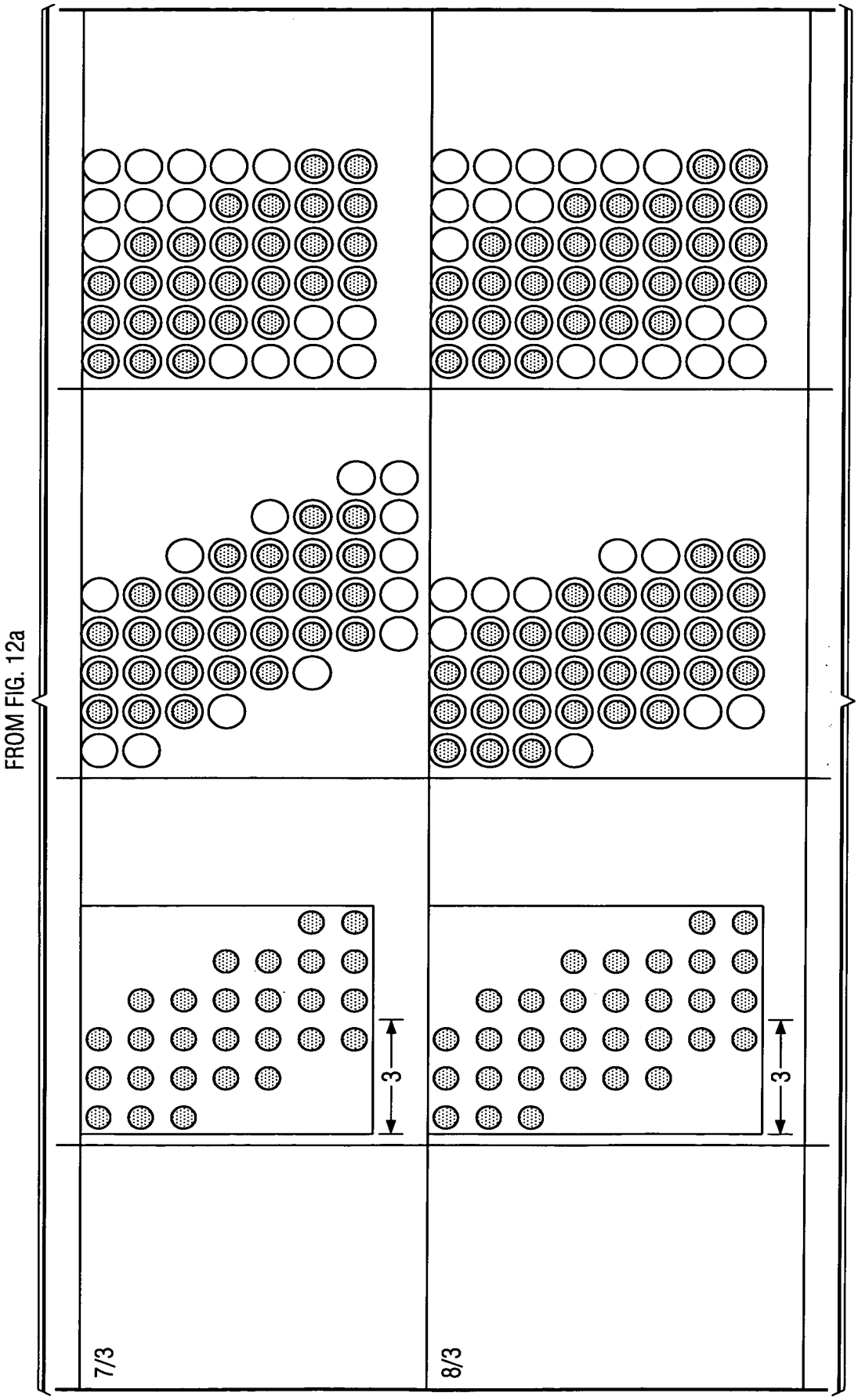


FIG. 12b

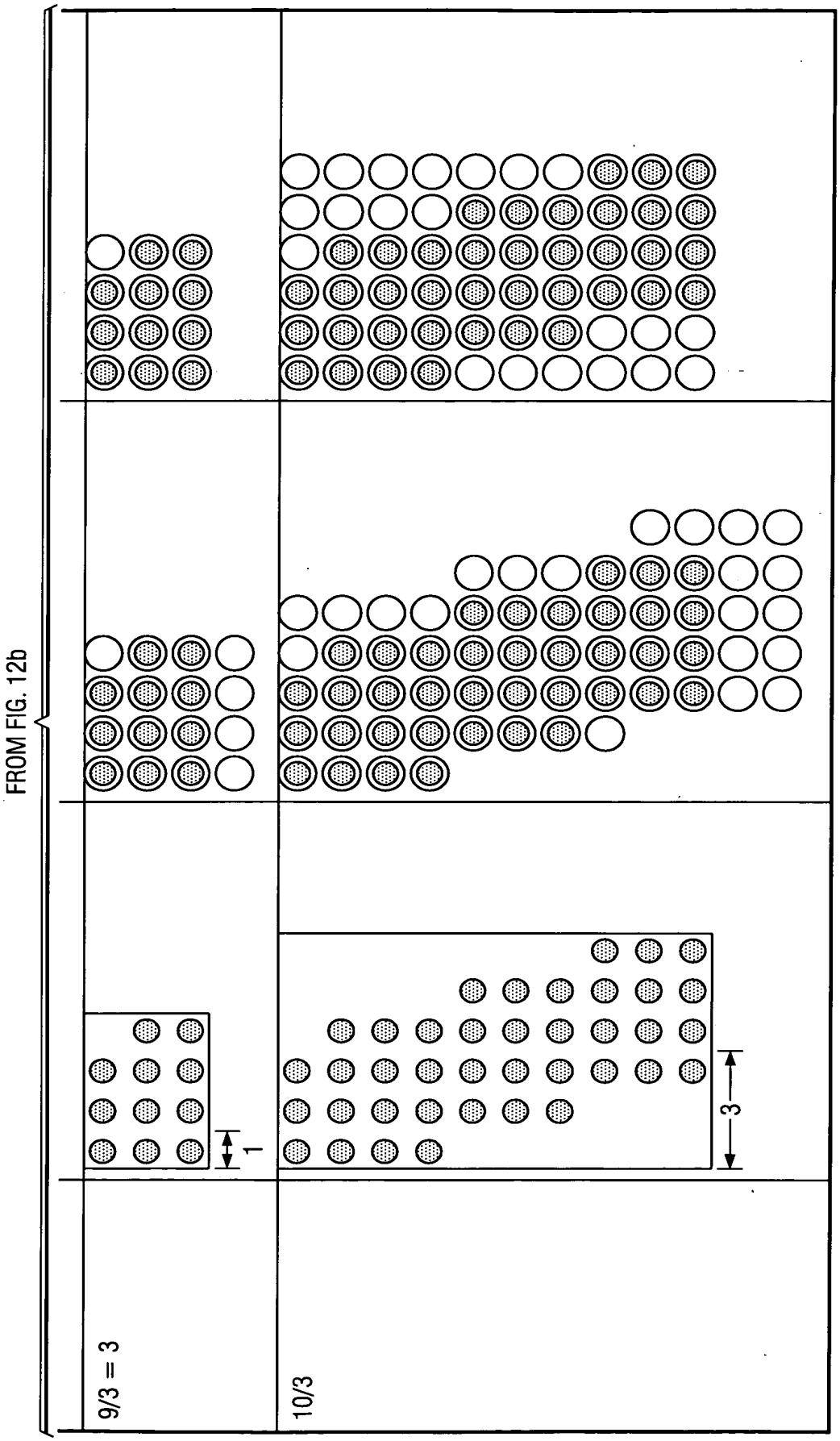


FIG. 12c

RESAMPLING FACTOR	Multiply_Factor	H_Arch_Kernel	FILTER TAPS	DATA STEP TO NEXT GROUP OF OUTPUTS	STARTING ACCESS DATA POINT
4/3 HORIZONTAL	1	1:1 SLOPE 4 OUTS	4	N/A	-2
VERTICAL			4	1	-2
5/3 HORIZONTAL	1	2:1 SLOPE 4 OUTS	4	2	-1
VERTICAL			5	1	-3
2/1 HORIZONTAL	2	2:1 SLOPE 4 OUTS	4	N/A	-1
VERTICAL			5	0	-1
7/3 HORIZONTAL	1	2:1 SLOPE 4 OUTS	5	2	-2
VERTICAL			6	0	-1
8/3 HORIZONTAL	1	4 TALL	5	1	-1
VERTICAL			6	0	-1
3/1 HORIZONTAL	1	4 TALL	4	N/A	-1
VERTICAL			4	0	-1
10/3 HORIZONTAL	1	4 TALL	5	1	-1
VERTICAL			6	0	-1

FIG. 13